Appendix K. Data Base Services

INTRODUCTION

The goal of the ECAT is to ensure portability, interoperability, scalability, compatibility, and extensibility among agencies and value-added networks (VANs). The proposed data bases must track all required elements in a centralized, decentralized, or combination of both methods. Agencies and trading partners must have access to government-wide acquisitions policies and procedures, partner agreements, and other pertinent information. Other data bases, discussed below, will provide services for agencies and trading partner: agency application system, gateway, and VANs.

The EC data base architecture may be implemented as client/server, distributed, or a combination of both environments, as depicted in Figure K-1. This offers portability, interoperability, and scalability for data base application at the source-code level. The approach adopted by this architecture is based on the National Institute of Standards and Technology (NIST) Open Systems Environments (OSE) standards-based framework.

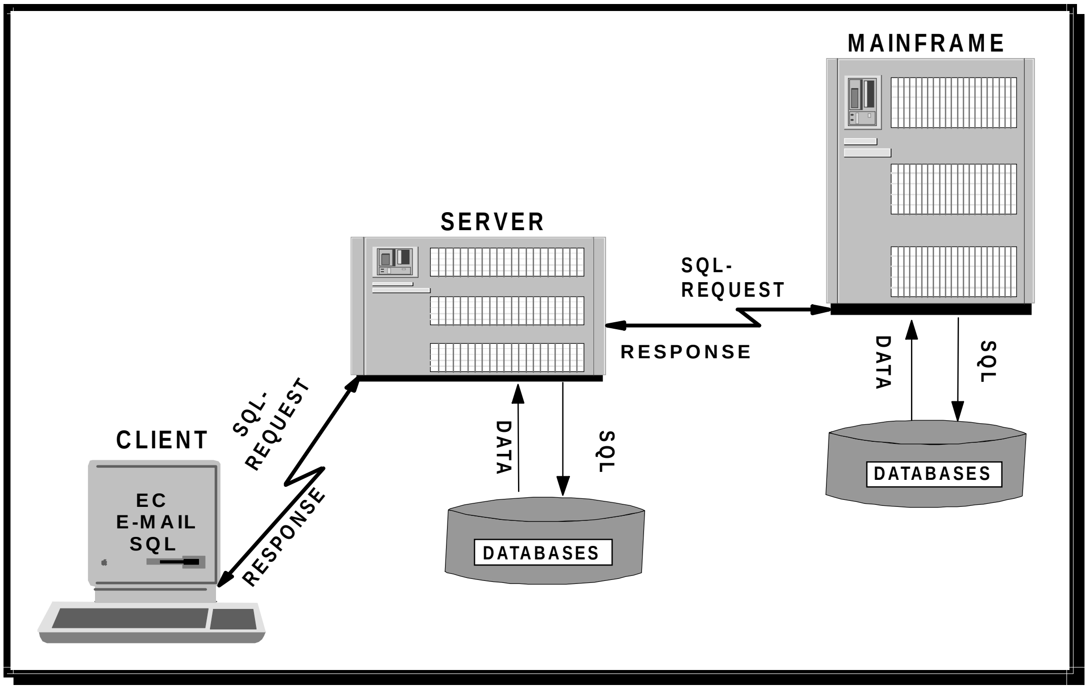

Figure K-1. Overview of EC/EMAIL/SQL Government-wide Client/Server Distributed Data Multiplier Environment

Access to data bases such as trading partner agreements, FAR, implementation conventions, and the trading partner registration data base, do not directly impact the capability of any of the above architectures to convey information between agencies and trading partners. Periodic access to these data bases will be useful in monitoring adherence to procurement regulations and implementation constraints. Access to the trading partner registration data base will allow agencies to check trading partner status before issuing contract awards.

With a centralized architecture, if the network entry point (NEP) (or any other common point though which all traffic passes) has access to all these data bases, then common regulations and constraints can be monitored (or enforced) at this one location. This same monitoring can be achieved with the distributed architectures, but it will have to done at either the gateway or agency. The impact of requiring monitoring at many locations may not be significant. In any case, agencies must maintain agency specific policies and procedures, trading partner agreements (which may vary from agency to agency), and procurement regulations, which will necessarily require enforcement at the agency location.

Other data bases will be used to facilitate communications between agencies and trading partners. From the perspective of agencies and trading partners, there are no significant differences on the required data base services that result from choosing a centralized or noncentralized architecture. Agencies and trading partners will both need access to these data bases to look up electronic mail addresses.

AGENCIES/TPS APPLICATION SYSTEMS

The procurement or financial users should have all applications behave the same way, with the same “look and feel.” The application system will provide the user with the capabilities of using EC to populate and access various data base systems. The application system also will provide structured query language (SQL) capabilities to create, access, update, and maintain various data base systems.

EC and SQL Application Programming Interfaces (APIs) for the end-user may include the use of embedded or stores procedure codes. This API approach will ensure that the application source code is portable, transparent, and independent of the underlining system architecture. (The SQL API concept is discussed in more detail below.)

EC and SQL will be used for various functions among agencies and between agencies and trading partners/VANs. Users must use FIPS 161, EDI (X12) transaction sets to populate data bases and FIPS 127-2, SQL to create, access, update, and maintain various common-shared data base (14+) systems. Users can use X.500 directory services with SQL to populate the data bases (i.e., trading partner). For example, trading partners will submit 838 registration application for X12.838 translation through the use of interagency APIs. The API will make the connection for the application to be translated into the X12 transaction set over the virtual network. The transaction will be placed into X.400 enveloped with X.435/F.435 security features. X.500 will be use to link with the SQL API for data caching, synchronization, and integrity to the data base management system (DBMS) engine (i.e., Informix, Oracle, Sybase) to populate, query, or update the trading partner registration data base.

GATEWAY/VAN SYSTEMS

The gateway and/or VAN will translate EC documents into FIPS-161, EDI (X12) compliant document. These documents will be transmitted over the virtual networks and may be placed into X.400 enveloped with security features (i.e., X.435/F.435). X.500 directory services will be used along with SQL to provide data base services.

DATA BASE MANAGEMENT SYSTEM (DBMS)

DBMS software will provide for management of data. The DBMS must comply with FIPS 127-2, (SQL2) and FIPS 151-2, Portable Operating Systems Interface (POSIX) native platform. This will ensure SQL applications are portable, interoperate, and scalable in a multivendor heterogeneous environment. The Data Dictionary/Repository Services must comply with FIPS 156, Information Resource Dictionary System (IRDS). One of the dictionary/repository features is to provide SQL with “data location” services whether in a client/server with multitier platforms, distributed, or combination of both environments. Remote Data Access (RDA) must comply with International Standards Organization (ISO) 9759 for distributed data. RDA should be use to access data whether in an homogenous or heterogeneous environment. However, the use of Open Systems Foundation (OSF) Distributed Computing Environment (DCE) should be considered for accessing and updating distributed data bases. We advise the “two-phase-commit” approach to update the data bases, but users should beware of data integrity when using this feature in heterogeneous environment. The DBMS security features must provide support for data integrity to all agencies/TPs for reading, writing, and updating procurement and financial data base systems.

SQL API CONCEPT

Agency and trading partner application programs must be able to communicate with the DBMS through sets of API calls. Typically, SQL API calls are used in a distributed client/server environment. In this configuration, the API code is located on the client system, where the program is executed. The DBMS program is located on the data base server system where data are stored. API call from the application takes place locally within the client (agency/TP) systems. Communication between the API and the DBMS takes place over the network (gateways/VANs). SQL API calls offer an advantage of minimizing the amount of network traffic between the APIs and the DBMS, particularly when APIs consist of “stored procedures” code, rather than embedded SQL code contained within an application program (written in “C”, COBOL, etc.).

To illustrate this point, compare two client-server implementations, one with embedded SQL interface and the other using SQL API and stored procedures code. Using the embedded approach, each SQL statement must be sent individually across the network from the client to the server; the client retrieves the results from the DBMS row-by-row. When using the SQL server API approach, with stored procedures code, a single SQL statement is sent from the client across the network to the DBMS. The results are a single “stream” of “query” messages. The single stored procedure API call over the network and back reduces the amount of traffic and greatly improves the network performance compared to the embedded SQL code.

Sample SQL API calls to access the DBMS basic operation are depicted in Figure K-2.

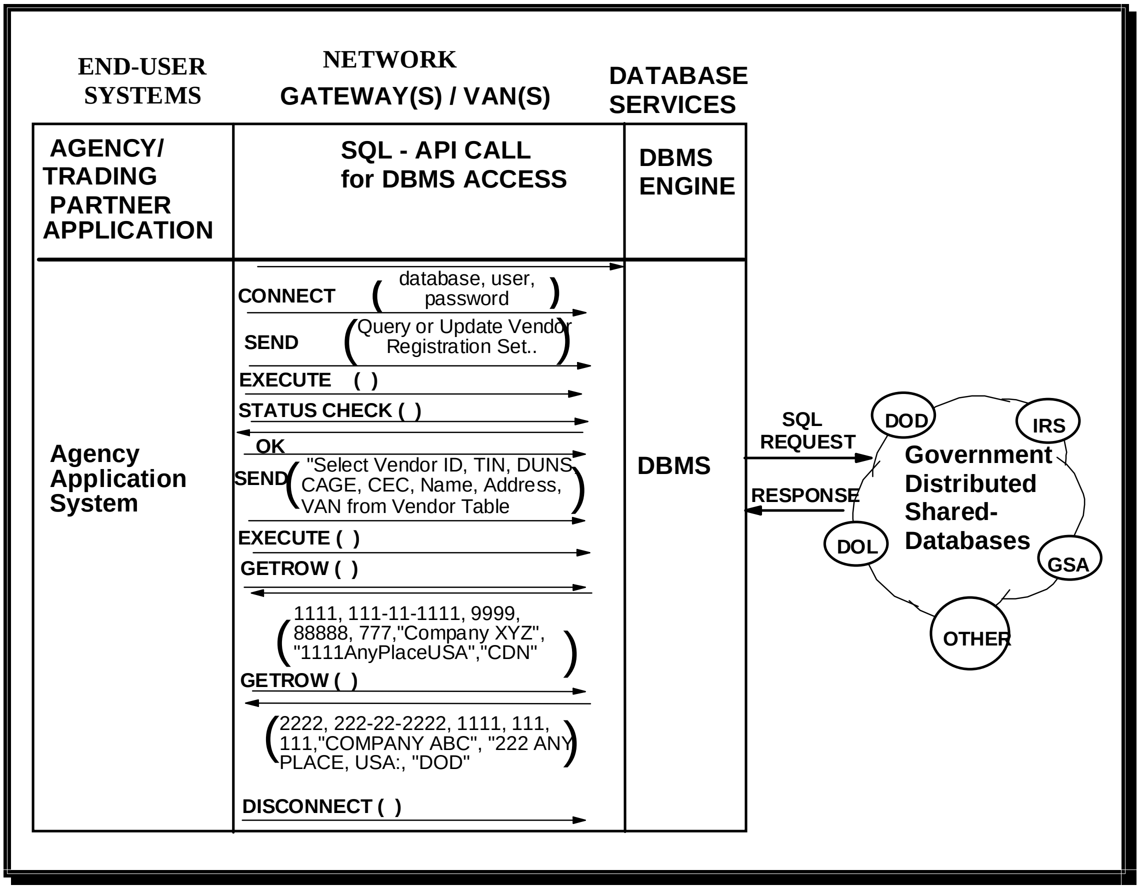
END-USER
SYSTEMS
NETWORK
GATEWAY(S) / VAN(S)
DATABASE
SERVICES
AGENCY/
TRADING
PARTNER
APPLICATION
SQL - API CALL
for DBMS ACCESS
DBMS
ENGINE
Agency
Application
System
CONNECT (database, user, password)
SEND (Query or Update Vendor Registration Set..)
EXECUTE ( )
STATUS CHECK ( )
OK
SEND ("Select Vendor ID, TIN, DUNS, CAGE, CEC, Name, Address, VAN from Vendor Table)
EXECUTE ( )
GETROW ( )
(1111, 111-11-1111, 9999, 88888, 777,"Company XYZ", "1111AnyPlaceUSA","CDN")
GETROW ( )
(2222, 222-22-2222, 1111, 111, 111,"COMPANY ABC", "222 ANY PLACE, USA:, "DOD")
DISCONNECT ( )
DBMS
SQL
REQUEST
RESPONSE
DOD
IRS
Government
Distributed
Shared-
Databases
DOL
GSA
OTHER

Figure K-2. Sample SQL API Calls to Access Government-wide Distributed Shared Data Bases

The following is a discussion of SQL API calls as they relate to Figure K-2:

- The application program starts to access the EC enterprise-wide shared data base with an SQL API call that CONNECT() the program to the DBMS and to specific data bases such as IRS, GSA, DOL, and DoD.

- The program sends SQL statements to the DBMS by building the SEND() statements as text strings in the buffer, then EXECUTE() API calls to pass the buffer contents to the DBMS.

- The program checks the status of the DBMS requests by making a STATUS_CHECK() API call, and if “OK” is not returned, then any errors are handled.

- The program uses the SEND(), EXECUTE(), and GETROW() to retrieve the results of SQL QUERY into the buffers.

- The program is terminated by making the DISCONNECT() API call to the DBMS.

EC DATA BASE ISSUES

The use of EC for acquisition requires support of a number of data bases as depicted in Figure K-3. The following data bases have been identified for this purpose: trading partner registration, trading partner agreements, government-wide Federal Acquisition Regulations (FAR), and agency-specific FAR supplements. In addition to these, data bases for EDI translation and for communication services are necessary. Issues related to these latter two data base systems are not specific to electronic commerce and thus are not addressed here. Issues related to the remaining data bases are more closely examined below.

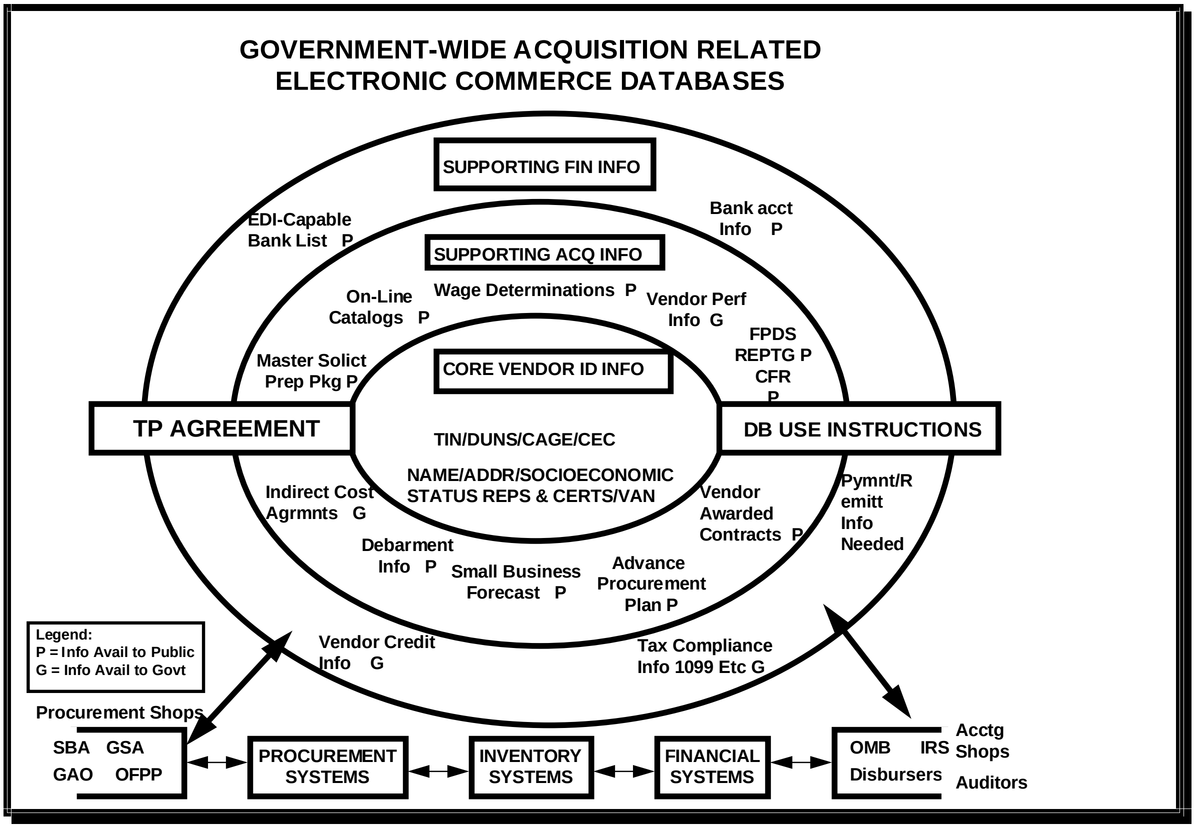

Figure K-3. Government-wide EC Data Bases

The trading partner registration process will be accomplished using the X12 838 transaction set. Trading partners will complete the 838 transaction set, and the information it contains, along with government provided information, will be stored in a data base. Since standardized EDI forms will be used, EDI translation services will be necessary. EDI translation services are also necessary for exchanging transaction sets used in conducting procurement electronically. In all of the architectural models presented for this purpose, EDI translation is provided by the VAN module. It would thus seem sensible to have access to this data base through the VAN. Following this approach, trading partners could employ whatever means they will otherwise use for submitting EDI forms to agencies, and register by electronically mailing the 838 form to a VAN. With access to the trading partner data base being available through the VAN, the VAN can translate the form and update the trading partner data base with the trading partner provided data.

Subsequent to populating the trading partner data base, the information stored there must be accessible to each government agency. Again, since each agency will have access to a gateway, if access to the trading partner data base is available through the gateway, it will also have access to the trading partner registration data base. This will leave the method for utilizing this data as a local issue, determined by each agency. Agencies could run local applications that access the data base for trading partner specific information before contracts are awarded, or anytime after a bid is received. Alternatively, applications could run SQL that automatically query the data base whenever a trading partner bid is received. Any trading partner information stored in the data base could then be supplied to the agency along with received bids. If trading partner specific information needs to accompany the bid, this information would be added at the gateway rather than the VAN.

The next question is the form the data bases should take and who should be responsible for their administration. For the data bases other than the vendor registration and trading partner data bases, the information they hold is determined by the government. Thus, the government is either directly, or through contract, responsible for their administration. Although the vendor data base contains information largely provided by vendors, the information will be used by procurement officials, and therefore the government should be responsible for administration. The trading partner data base is for holding agreements jointly arrived at by both government and vendor, and thus administrative responsibility may have to be decided upon by both agencies and vendors.

As of now, the structure of the information stored in the guidelines and trading partner data base is yet to be determined. If the information they contain is relatively unstructured (such as the FAR and FIRMR text), then a text data base is the most suitable choice. If the data is more structured, such as will be the case for the vendor registration data bases, a relational data base system is preferred since it has a solid mathematical foundation.

The FAR and FIRMR data bases will be necessary when issues regarding regulations arise, the trading partner data base will be needed for clarifying arrangements between agencies and vendors. The guidelines data base will be necessary for understanding how and what EDI forms are used. Although access to these data bases will be important, access to them will not routinely be necessary in conducting procurement activities. This is not the case with the vendor data base. Routine access to the vendor data base will be required before procurement officials can make final determination on awards issuance. Thus the primary focus below is on the vendor data base.

In the previous section it was argued that X.500 was the appropriate choice for storing both vendor and agency communications information. The major difference to consider in determining the best choice for the vendor data base is in the type of data to be stored. Generally, relational data base systems are preferred for storing structured data. They have a solid mathematical foundation, offer better data manipulation capabilities than X.500, and use a widely implemented standard language for access, SQL.

There are a number of options for how relational data bases might be used. There could be a relational data base that is replicated as necessary, or there could be many copies of the same data base with no single one of the copies serving as the master copy. Alternatively, the data base could be distributed (i.e., all copies would have the same data base structure) but contents could vary from one geographical area to another to satisfy local requirements. One way of achieving this is by developing an application that uses X.500 to track the distribution of the data. X.500 could be combined with RDA, the remote data base access protocol, to support data distribution using relational data base systems.

Interaction with remote relational data base systems is achieved through the use of RDA. When RDA is used to access a remote data base, if the desired data is stored at the remote site, it is returned to the user through the RDA protocol. If the desired data is not located at the site accessed, rather than the user searching for it elsewhere, an application program could use X.500 to locate the data and RDA to return the data to the user in the same way it was returned in the first case. From this perspective, the distribution of the data would be rendered transparent to the user. From a user's perspective it makes no difference at which geographical site the data base is entered, the results of any operation are the same regardless (discounting access controls). The data base in effect maintains two views of the data. One is the logical view of the data that is presented by SQL and seen by the user. The other is the physical view of the data, representing the actual physical distribution of the data, maintained by X.500. Combining X.500 and relational data bases allows a single integrated view of the data to be maintained and renders the distribution of the data transparent to the user. It also allows new sites for managing the data to be added indefinitely without impacting the user's logical view of the data. Combining X.500 and relational data bases then provides an attractive solution that exploits the strengths of each: X.500 for tracking data, relational data bases for storage and manipulation of data.

Agencies will need to decide whether frequently used data bases could be replicated on-site if the requirements for updating were reasonable (e.g., updating a small portion of the data every 24 hours).

Agencies will need to decide whether a frequentlyused subset of a data base could be replicated on-site, and the remainder of the data base could be located at a VAN, NEP, or other central site.

Agencies will need to decide whether a frequently updated data base could reside at a VAN, NEP, or other central site.

Agencies will need to decide whether a VAN, or NEP could add data base information to a transaction, so the recipient would not need to access the data base (for example, adding information from the trading partner registration data base to the trading partner's quote or blocking a quote from an ineligible, suspended, and debarred trading partner).

RECOMMENDED GOVERNMENT-WIDE COMMON-SHARED DATA BASES

Common data base types are shown in Table K-1.

Table K-1. Data Base Types by Source

Data Base Types	Agency Sources
Tax compliance	Treasury/Internal Revenue Service (IRS)
Vendor registration	Department of Defense (DOD) and other agencies
Debarred, performance, trading partner agreements, master government contracts, on-line catalogs, FAR/Federal Information Resource Mangement Resource (FIRMR)	General Service Administration (GSA)
Wage determination	Department of Labor (DOL)
Indirect cost agreements	Defense Contract Audit Agency (DCAA)
Master solicitation documents	Health and Human Services (HHS)
Small business forecast	Agencies
Advance procurement plan	Agencies
Term and conditions for small purchases	GSA
Acess to CFRs	Government Printing Office (GPO)
Payment offset information	Treasury
1099 data	Treasury
EC transaction into FPDS	GSA
EDI remittance compliant banks list	Treasury
Master payment and tax due offset file	Treasury
Vendor past performance file	DOD

AGENCIES APPLICATION SYSTEMS

Agencies need to develop DBMS capabilities for the following:

- Provide access to the internal agency management information, acquisition, and financial systems.
- Archive electronic file of documents, information transactions, and other pertinent procurement and financial information.

- Access and store data responses (i.e., message confirmation).

GATEWAYS

Agencies need to develop DBMS capabilities for the following:

- Provide archive and audit trails for electronic file of documents, information transactions, and other procurement and financial information.
- Provide security for sensitive, classified, business data, data integrity, multi-Federal agency and trading partner access control through password authentication methodology, encryption and decryption, and key management elements including records created through password log-on and modifications.

VALUE-ADDED NETWORKS

VANs must have the following DBMS capabilities:

- Provide archive and audit trails for electronic file of documents, information transactions, and other procurement and financial information.
- Provide security for sensitive, classified, business data, data integrity, multi-federal agency and trading partner access control through password authentication methodology, encryption and decryption, and key management elements including records created through password log-on and modifications.

SUMMARY

DBMS services must provide support to the acquisition objectives of the presidential memorandum. The DBMS for the EC system architecture will be based on the NIST standards-based OSE framework (see Figure K-4). We advise that some of the proposed open systems standards may have limited functionality because of the degree of stability or incompleteness of the standards. Therefore, where there are no stable de jure (i.e., FIPS, ISO, ANSI) standards, at times we may select de facto (i.e., DCE) standards or industry GAP filler products that are based on the OSE concept for implementing EC and other government projects. The following solutions are for implementing EC data base architecture government-wide now and is flexible and general enough to allow agencies to include stable open systems standards when they become available in the future:

- Adopt distributed data, client/server architecture for EC data base.
- Adopt SQL as the data base technology.
- Build or rebuild existing data bases to comply with NIST OSE standards as depicted below:
    - Agencies and trading partners with or without access to VANs that want to do EC government-wide acquisitions. The VANs must select a DBMS that implements the logical data view of the EC system architecture objective configuration. This is also true for trading partners without access to VANs.
    - Agencies and trading partners with or without access to VANs that want to do EC government-wide acquisitions. The VANs must select a data dictionary/directory service that conforms to the requirements in FIPS 156, IRDS. This is also true for trading partners without access to VANs.
    - Agencies and trading partners with or without access to VANs that want to do EC government-wide acquisitions. The VANs must select a DBMS that conforms to the requirements in FIPS 127-2, “Database Language SQL.” In addition, conformance is required 18 months after FIPS 127-3, “Database Language SQL,” becomes available. This is also true for trading partners without access to VANs.
    - Agencies and trading partners with or without access to VANs that want to do EC government-wide acquisitions. The VANs must select distributed data base services that implement the requirements defined in “Remote Database Access (RDA),” ISO Draft Proposal (DP) 9759; “Part 1: Generic Model Service and Protocol,” Joint Technical Committee (JTC1)/Special Committee (SC) 21N4282; and “Part 2: SQL Specialization,” JTC1/SC21N4281. This is also true for trading partners without access to VANs.
    - Agencies and trading partners with or without access to VANs that want to do EC government-wide acquisitions. The VANs must select a DBMS that will comply with data management security. This is also true for trading partners without access to VANs.

- Acquire DBMS COTS products that conform with the FIPS 127-2, SQL and FIPS 151-2, POSIX standards “now.” Also, other related FIPS and ISO data base standards are discussed above.
- Acquire X.500 COTS to link with relational DBMS for EC transaction sets (i.e., 838) to populate the trading partner registration data base. We advise this approach only if EC is used for data base population. Alternatively, agencies/TPs/VANs can use SQL as a generaK-purpose solution to populate, query, and update data bases.
- Acquire COTS or develop interagency standardized API interfaces for front-end middleware (or enablers) and back-end data bases access.
- Develop application software as a short-term technology for remote query and update.
- Develop a prototype and evaluate the feasibility of RDA as a long-term technology for remote query and update.

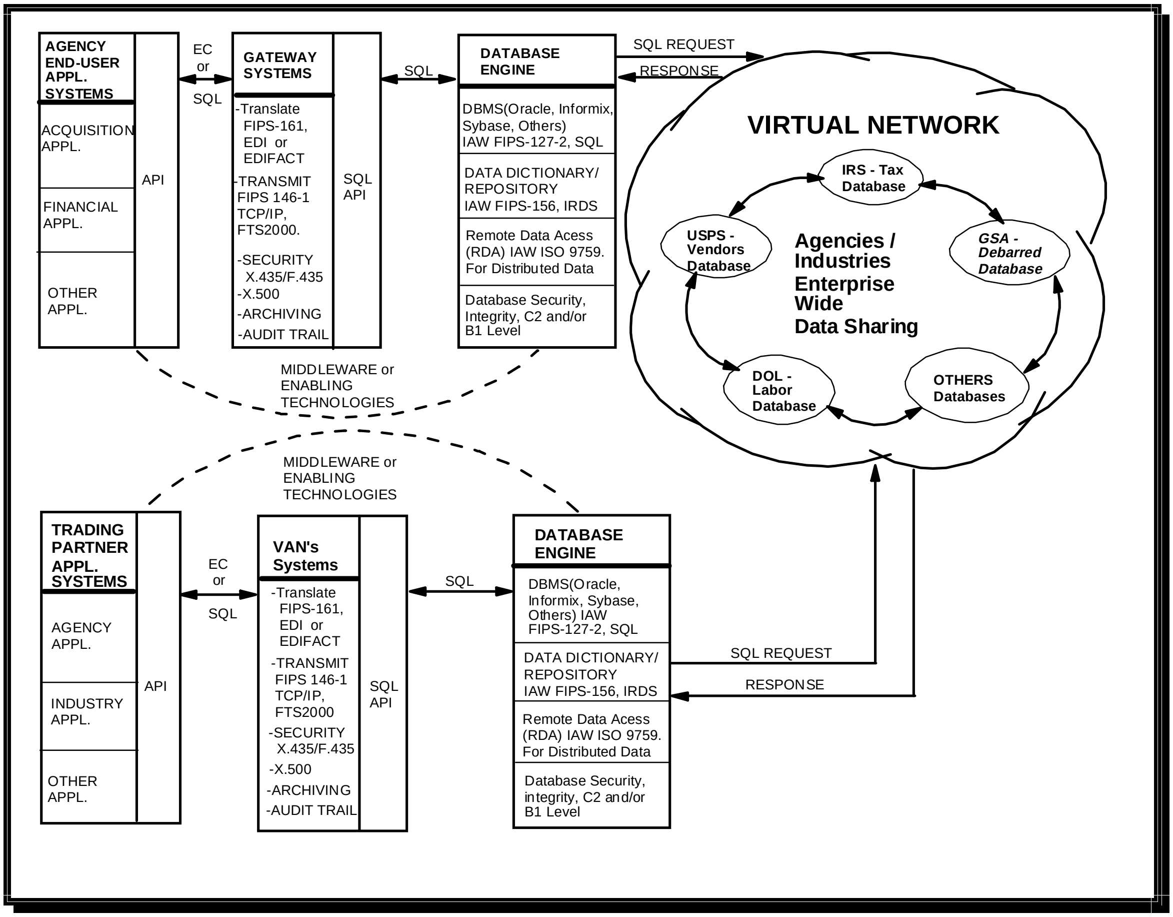
AGENCY END-USER APPL. SYSTEMS
ACQUISITION APPL.
FINANCIAL APPL.
OTHER APPL.
API
EC or SQL
GATEWAY SYSTEMS
-Translate FIPS-161, EDI or EDIFACT
-TRANSMIT FIPS 146-1 TCP/IP, FTS2000.
-SECURITY X.435/F.435
-X.500
-ARCHIVING
-AUDIT TRAIL
SQL API
SQL
DATABASE ENGINE
DBMS(Oracle, Informix, Sybase, Others) IAW FIPS-127-2, SQL
DATA DICTIONARY/ REPOSITORY IAW FIPS-156, IRDS
Remote Data Acess (RDA) IAW ISO 9759. For Distributed Data
Database Security, Integrity, C2 and/or B1 Level
SQL REQUEST
RESPONSE
VIRTUAL NETWORK
IRS - Tax Database
USPS - Vendors Database
GSA - Debarred Database
Agencies / Industries Enterprise Wide Data Sharing
DOL - Labor Database
OTHERS Databases
MIDDLEWARE or ENABLING TECHNOLOGIES
MIDDLEWARE or ENABLING TECHNOLOGIES
TRADING PARTNER APPL. SYSTEMS
AGENCY APPL.
INDUSTRY APPL.
OTHER APPL.
API
EC or SQL
VAN's Systems
-Translate FIPS-161, EDI or EDIFACT
-TRANSMIT FIPS 146-1 TCP/IP, FTS2000
-SECURITY X.435/F.435
-X.500
-ARCHIVING
-AUDIT TRAIL
SQL API
SQL
DATABASE ENGINE
DBMS(Oracle, Informix, Sybase, Others) IAW FIPS-127-2, SQL
DATA DICTIONARY/ REPOSITORY IAW FIPS-156, IRDS
Remote Data Acess (RDA) IAW ISO 9759. For Distributed Data
Database Security, integrity, C2 and/or B1 Level
SQL REQUEST
RESPONSE

Figure K-4. OSE Standards-Based Data Base Architecture